



## System Architektur (Architecture)

### Zwischenklausur (Voluntary Examination)

**WS 2000/2001, 8. Januar 2000**

- Bitte tragen Sie zuerst auf allen Klausurblättern Ihren Namen, Vornamen und Ihre Matrikelnummer ein, auch auf den Konzeptblättern. *Please enter your last name, first name and matriculation number on each page (including used and unused draft pages).*
- Die Prüfung dauert 60 Minuten und besteht aus 5 Aufgaben auf 11 Seiten und zwei Konzeptblättern. *You have 60 minutes to complete your answers. The examination consists of 5 questions on 11 pages. You have received two additional blank pages for drafts, etc.*
- ~~Die Prüfung ist mit mindestens 20 Punkten von 60 erreichbaren Punkten bestanden. *You pass the examination by obtaining at least 20 marks out of the possible 60 marks.*~~
- Es sind keinerlei Hilfen erlaubt! *No additional means are allowed!*
- Die Prüfung gilt als nicht bestanden, wenn Sie versuchen, aktiv oder passiv zu betrügen. *You fail the examination if you try to cheat actively or passively.*
- Wenn Sie zusätzliches Konzeptpapier benötigen, verständigen Sie bitte die Klausuraufsicht. *If you need more draft pages please notify one of the supervisors.*
- Bitte machen Sie eindeutig klar, was Ihre endgültige Lösung zu den jeweiligen Teilaufgaben ist. Teilaufgaben mit mehreren Lösungen oder mit widersprüchlichen Teilen werden mit 0 Punkten bewertet. *Make sure that it is absolutely clear what your final solution is for each subquestion. Subquestions with multiple solutions or with contradicting parts are void: 0 marks.*

**Die folgende Tabelle wird von uns ergänzt! *The below table is completed by us!***

Aufgabe/ <i>Question</i>	1	2	3	4	5	Total
Erreichbare Punkte <i>Possible marks</i>	12	12	12	12	12	60
Erreichte Punkte/ <i>Obtained marks</i>						
Note/ <i>Grade:</i>						

Nachname/ <i>Last name</i>	Vorname/ <i>First name</i>	Matrikelnummer/ <i>Matriculation number</i>

**Aufgabe/Question 1 (Zum Aufwärmen/Warm up, 3 + 3 + 1 + ... + 1 Punkte/marks)**

1. „Zählen Sie die drei gebräuchlichen Kriterien auf, welche von praktikablen Seiteneretzungsverfahren (*page replacement*) benutzt werden, und geben Sie hierzu jeweils ein konkretes Seiteneretzungsverfahren an!“

*“Enumerate the three commonly used criteria used by practical page replacement algorithms, and give a concrete paging algorithm for each of these criteria.”*

a) .....

.....

b) .....

.....

c) .....

.....

2. „Zählen Sie drei typische Kernschnittstellenoperationen (*kernel interface function*) auf!“

*“Enumerate at least three typical kernel interface functions.”*

a) .....

b) .....

c) .....

Einige der folgenden Aussagen sind korrekt, einige inkorrekt. Unterstreichen Sie „korrekt“, wenn die Aussage korrekt ist, unterstreichen Sie „inkorrekt“, wenn die Aussage inkorrekt ist.

*Some of the following statements are correct, some are incorrect. Underline “korrekt” if the statement is correct; underline “inkorrekt” if the statement is incorrect!*

3. „In einem Einprozessorsystem, in dem bis zu 1000 Threads gleichzeitig aktiviert sind, ist es effizienter, den Threadzustand „bereit“ (ready) durch eine explizite Datenstruktur zu implementieren als nur durch ein entsprechendes TCB-Attribut.“

*“In a single-processor system with up to 1000 currently activated threads, it is more efficient to implement the thread-state “ready” by a separate data structure than by a corresponding TCB attribute.”*

korrekt

inkorrekt



<b>Nachname/Last name</b>	<b>Vorname/First name</b>	<b>Matrikelnummer/ Matriculation number</b>

**Aufgabe/Question 2****(6 + 4 + 2 Punkte/marks)**

1. „Ergänzen Sie die folgende Programmskizze so, daß eine Implementierung für eine generalisierte Synchronisation für **n>2 Threads** entsteht! Gehen Sie davon aus, daß die Kernschnittstellenoperation `n_synchronize` atomar ist.“

“Complete the missing lines in the following program template to implement a generalized synchronization module for  $n > 2$  threads. Assume the kernel interface operation `n_synchronize` is atomic.”

```

module generalized synchronization
  export n_synchronize
  import UnblockThread, BlockThread
  type n_sync = record

      W: waiting thread := empty    {waiting queue}
  end

  procedure n_synchronize(SY:sync)
    begin

      end n_synchronize
end module.

```

**Fortsetzung von Aufgabe 2 / Question 2 continued:**

**(4 + 2 Punkte/marks)**

2. „Geben Sie in Form einer knappen Programmskizze ein Anwendungsbeispiel für obiges generalisierte Synchronisationsmodul an!“  
*“Try to illustrate in a brief program template an example application of the above generalized synchronization module.”*

3. „Welche Hardwareunterstützung würden Sie verwenden, um ein **atomares** `n_synchronize` in einem Ein- bzw. Mehrprozessorsystem zu implementieren?“  
*“Which hardware support would you use to implement an atomic `n_synchronize` in a single- and in a multi-processor system?”*

**Single-processor system:**

.....  
.....

**Multi-processor system:**

.....  
.....



**Fortsetzung von Aufgabe 3 / Question 3 continued: (3 + 3 + 2 Punkte/marks)**

2. „Gegeben seien zwei nebenläufige Transaktionen  $T_1$  und  $T_2$ , von denen jede aus eine Reihe von Lese- bzw. Schreiboperationen auf eine gemeinsame Datenbank aufgebaut sei. Die Datenbank bestehe aus  $d \gg 1$  Datenobjekten  $do_i$ , von denen jedes separat durch einen Read\_Lock bzw. Write\_Lock vor gleichzeitigem Zugriff geschützt werden kann. Welche Bedingungen müssen gegeben sein, damit obige Transaktionen im Konflikt stehen?“

*“Assume there are two concurrent transactions  $T_1$  and  $T_2$  each of which consists of a sequence of read- and write-operations to a shared database. This database consists of  $d \gg 1$  data objects each of which can be protected against concurrent accesses by a separate Read\_Lock and/or by a separate Write\_Lock. Which conditions may lead to conflicting transactions?”*

.....

.....

.....

.....

3. „Welche der vier Eigenschaften des ACID-Prinzips wird durch einen serialisierbaren (*serializable*) Ablaufplan herbeigeführt?“

*“Which of the four requirements of the ACID-principle is achieved by a serializable schedule?”*

.....

4. „Was sind die Vor- und Nachteile, wenn in einer gemeinsamen Datenbank jedes Datenobjekt separate Sperren (*locks*) besitzt?“

*“What are the pros and cons of having separate locks per data object in a shared data base?”*

.....

.....

.....

.....

.....



**Fortsetzung von Aufgabe 4 / Question 4 continued:****(4 Punkte/marks)**

3. „Gegeben sei ein Seitenreferenzstring für einen Applikationsthread mit einer Hauptspeichermenge (*resident set*) von  $m$  Kacheln, die initial alle leer seien. Der Seitenreferenzstring habe die Länge  $p$ , wobei  $n$  verschiedene Seitennummern enthalten sind. Wie lauten für jeden Seitenersetzungsalgorithmus

- a) die untere Schranke für die Seitenfehleranzahl,
- b) die obere Schranke für die Seitenfehleranzahl?“

“Consider a page reference string for an application thread with a resident set of  $m$  frames, initially all empty. The page reference string is of length  $p$  with  $n$  distinct page numbers in it. For any page replacement algorithm,

- a) what is the lower bound on the number of page faults and
- b) what is the upper bound on the number of page faults.”

a) .....

b) .....

4. „Skizzieren oder beschreiben Sie die Wirkungsweise von *Copy-on-Write*! In welchem Zusammenhang wird *Copy-on-Write* eingesetzt?“

“Describe the method *Copy-on-Write* method. In what context is *Copy-on-Write* used?“



**Fortsetzung von Aufgabe 5 / Question 5 continued:****(5 Punkte/marks)**

2. „Gegeben sei ein symmetrisches Multiprozessorsystem (*SMP*) aus 4 Prozessoren, die alle über den Systembus an den gemeinsamen Hauptspeicher (*main memory*) angeschlossen sind. Pro Prozessor existiere ein L1- und ein L2-Pufferspeicher (*cache*). Auf Prozessor  $ZP_4$  läuft ein Schachprogramm mit einer großen Arbeitsmenge (*working set*), die nicht vollständig in den L2-Cache paßt. Dieses Schachprogramm läuft unabhängig von allen sonstigen Anwendungen. Drei kooperierende Threads mit jeweils mittelgroßer Arbeitsmenge (*working set*), d.h. jede von ihnen paßt vollständig in einen L2-Cache, laufen echt parallel auf den Prozessoren  $ZP_1$  bis  $ZP_3$ . Diese drei kooperierenden Threads synchronisieren sich mittels des unten skizzierten „**Spinlocks**“:

Warum läuft das Schachprogramm deutlich langsamer als auf einem Einprozessorsystem?“

“Given a symmetric multi-processors system (*SMP*) consisting of 4 processors operating on a common main memory via a system-bus. L1 and L2 caches exist per processor. Processor  $ZP_4$  executes a chess program with a large working set that in general does not fit into its L2 cache. This chess program runs independently of all other applications. Three cooperating threads with medium sized working sets, i.e. each working set fits into an L2-cache run in parallel on processors  $ZP_1$  through  $ZP_3$ . The three cooperating threads synchronize with each other by means of the following **spinlock**:

Why does the chess program run substantially slower than on a single-processor system?“

DO

```

reg := MyThreadId ;
xchg (SpinLock,reg) {special atomic exchange operation}
                    {exchanges mem/cache variable SpinLock and reg}
UNTIL reg = 0 OD;
...
... critical section ...
...

SpinLock := 0

{SpinLock = 1 ⇔ another Thread is in its critical section}
{SpinLock = 0 ⇔ critical section is free }

```

.....

.....

.....

.....

.....

<b>Nachname/<i>Last name</i></b>	<b>Vorname/<i>First name</i></b>	<b>Matrikelnummer/ <i>Matriculation number</i></b>

<b>Nachname/<i>Last name</i></b>	<b>Vorname/<i>First name</i></b>	<b>Matrikelnummer/ <i>Matriculation number</i></b>

<b>Nachname/<i>Last name</i></b>	<b>Vorname/<i>First name</i></b>	<b>Matrikelnummer/ <i>Matriculation number</i></b>

**Einverständnis/Agreement:**

Ich stimme zu, daß mein Zwischenklausurergebnis in einer durch ein Passwort geschützten Ergebnisliste im Web bekanntgegeben wird. Die Liste wird Einträge der Form: **Matrikelnummer, erzielte Punkte ~~und Note~~** enthalten.

*I agree that my results will be published on the web in a password protected web page. The web page will contain entries of the form: **matriculation number, obtained marks, and grade.***

Unterschrift/*Signature*: .....

-----